ADDRESS WRAP FUNCTION FOR ADDRESSABLE MEMORY DEVICES

Reference is made to Provisional Application Serial Number 60/104,889 Filed 10/20/98 which is incorporated herein.

Field of the invention

The invention relates to the providing of a wrap function useable in connection with an addressable random access memory that will wrap the address portion of data for additional use without disturbing the contents of the memory element and in particular to a special command or function capability that will selectably bypass the storage part of the memory assembly and pass the address portion on for use in collateral applications.

Background

In many data processing operations the information being processed is in the form of increments or words that carry an accumulation portion that is to be stored and an identification portion that tells the memory array where to put it. As progress in the art has taken place the arrays have become huge, the housekeeping and control circuitry very complex and the stored information both large and valuable. There have long been problems with such arrays in the ever increasing speed and size environment with testing, with timing and with component drift as examples. Where the memory array is made up of addressable elements assembled into semiconductor integrated circuitry, accessability and operational constraints severely limit quality and reliability maintenance testing and the adding of further capabilities.

As progress proceeds in the art it is expected that in the future dynamic addressable arrays of devices will utilize signaling techniques that require adjustments, in the verification that address information is being properly received, and in the timing of when data is launched or captured. It will further be desirable to periodically

re-initialize timing as components drift or where addressable devices, on coming out of a long period of inactivity, may be found to have undergone a timing change due to temperature or voltage variation.

A need is developing in the art to be able to maintain and to reestablish conditions at individual memory elements in an array without tampering with the information stored in the array.

Summary of the invention

The invention is a selectable function that permits the address portion of data words to be separated from the storable content portion and that address portion to be used for different purposes without disturbing the stored contents in the memory array. The invention may be viewed as a command capability that permits analysis of signals for errors in such items as addresses, impedance calibration, timing, and component drift that develop in and between regions of an overall memory array.

A technique of testing addressing is advanced whereby information on the address bus of a memory element such as a semiconductor integrated element is routed directly to the data pins of the element and driven back to the controller of the element, so that by comparing the information that the controller sent on the address lines, to the information received on the data lines, it can be determined if there are any fails in the address or data lines, without disturbing the stored contents of the memory array.

Techniques are advanced involving data responsive selectable array circuitry modification whereby the storage portion of the array is isolated and the address portion of the data is rerouted and refunctioned for such operations as timing, verification and component drift correction purposes.

The principles are illustrated with memory systems built of Synchronous Dynamic Random Access Memory with Double Data Rate (SDRAM-DDR) elements.

Brief Description of the Drawings

Figures 1 - 9 illustrate the application of the principles of the invention to the analysis of the integrity of address signals as they are propagated through an addressable memory array; wherein: Figures 1 - 6 illustrate the assignment of reference numerals pertaining to the invention to selected nodes of a typical standard in the art addressable memory system, in which:

Figure 1 illustrates a basic addressable memory element such as a semiconductor integrated circuit, in a typical package such as a dual in line package, with reference numerals assigned to pins and terminals that will be involved in the illustration of the invention.

Figure 2 illustrates the interconnected functional elements of a, typical in the art, Double Data Rate Synchronous Dynamic Random Access Memory (SDRAM-DDR), with reference numerals assigned to elements pertaining to the invention.

Figure 3 is an illustration of a function truth table for a typical in the art SDRAM-DDR, such as illustrated in Fig. 2.

Figure 4 is an illustration of a layout on a subassembly member or card of a two memory bank SDRAM-DDR of the type illustrated in Figs. 1 - 3 with reference numerals added pertaining to this invention.

Figure 5 is an illustration of a schematic diagram of a typical in the art two memory bank SDRAM - DDR of the type illustrated in Figs 1 - 4 with reference numerals added pertaining to this invention.

Figure 6 is a perspective illustration of the arrangement of the SDRAM-DDR type memory cards in a typical in the art computer system as illustrated in Figs 1 - 5 with reference numerals added pertaining to this invention, and in which

Figures 7 - 9 are structure, flow and timing diagrams illustrating the by pass implementation of the invention, wherein,

Figure 7 illustrates a bypass circuitry capability within the functional diagram of Fig. 2 conveying address and data port information with reference numerals pertaining to the invention.

Figure 8 illustrates the information flow of the invention in a computer system such as that of Fig. 6 with reference numerals pertaining to the invention, and,

Figure 9 illustrates in a timing diagram the performance when the system is in the wrap function or echo function mode of the invention with reference numerals pertaining to the invention.

Figures 10 - 13 illustrate the application of the principles of the invention to the control of the impedance of an Off Chip Driver, standard component used in the art in Dynamic Random Access Memory (DRAM) assemblies; wherein:

Figure 10 illustrates a typical DRAM data path through the assembly for writing data.

Figure 11 illustrates one arrangement of adjustment additions to a writing typical data path such as is shown in Figure 10 in order to implement the principles of the invention.

Figure 12 is a timing chart illustrating the effect of the adjustment additions of Figure 11.

Figure 13 illustrates an alternate arrangement of adjustment additions to a typical data path such as is shown in Fig. 10 in implementing the principles of the invention.

Figures 14 -16 illustrate the application of the principles of the invention in a rerouting implementation to the evaluation and control of timing in DRAM memory assemblies, wherein:

Figure 14 is a block diagram illustrating a typical timing control data path in a DRAM assembly.

Figure 15 illustrates an arrangement of adjustment additions to a data path of the type shown in Fig. 14 in the implementation of the principles of the invention in timing calibration., and,

Figure 16 is a timing chart illustrating the conditions produced in timing where adjustment additions are made in a typical data path such as is illustrated in Fig. 15.

Description of the invention

The selectable function of the invention that permits the address portion of data to be separated from the storable content portion so that that address portion can be used for different purposes, can be implemented in many ways such as through hardware additions, software instructions and combinations thereof. The implementations fall into groups where the storage arrays are by passed and groups where the storage arrays are isolated and the address data is rerouted and used for other purposes. In both types of groups the memory content remains undisturbed. The invention may be viewed as being a command or wrap function that permits analysis , verification and correction of variations in such data paths as address, timing, impedance variation and component drift in addressable memory assemblies without disturbing the content of the actual stored data in the memory assembly. The memory assembly is made up of an interrelated arrangement of storage and control entities. In Figures 1 -6 there is illustrated a typical Dynamic Random Access Memory (DRAM) made up of components and data paths and controls assembled on standard in the art packaging on cards and boards. This invention is directed toward providing a capability for analyzing critical types of data paths and providing correction so as to keep the interrelationship within proper limits without disturbing data that may be stored in the memory.

Referring to Figures 1 to 6, in Fig. 1 there is shown a basic addressable memory element such as a semiconductor integrated circuit, in a typical package such as a dual in line package, with reference numerals assigned to pins and terminals that will be involved in the illustration of the invention. In Fig. 1 the semiconductor integrated circuit element is labelled 101 and all signal pins are listed. There are address pins A0-A12, labelled 102, for address ing the memory contents by row and column. There are banks of select pins BA0-BA1, labelled 103, for addressing one of the 4 internal memory banks, command pins RAS, CAS, WE, and CS , labelled 104, which respectively refer to row address, column address, write enable, and chip select. There is a differential clock CLK CLK/ pair, labelled 105, for synchronizing operations within the chip to a system clock, and a clock enable pin CLE, labelled 106, for enabling and disabling the clock. There are 4, 8, or 16 data ports, shown in Fig. 1 as DQ0-DQ15 depending on the data width of the chip, with data strobes UDQS, LDQS, labelled 107, one for each 8 data bits. The data ports are used to send READ information data, or receive WRITE information data. The strobe is driven with the data on a read and receive operation, serving the function of a clock, with data present on both rising and falling edges of the strobe pulse. There are also voltage supply pins VDD and VSS for the internal circuitry, and I/O voltage supply pins VDDQ and VSSQ for the data and strobe pins, and a reference voltage Vref pin for receiving data.

Figure 1 and Figure 2 together illustrate the interconnected functional elements of a, typical in the art, Double Data Rate Synchronous Dynamic Random Access Memory (SDRAM-DDR), with reference numerals assigned to elements pertaining to the invention and together describe an addressable semiconductor device with an address and data port, which are essential data path locations in analysis of address accuracy.

In Figure 2, a functional diagram is shown with connections to the major parts of the Random Access Memory (RAM) device. In figure 2 , the clock is labelled (202), it is used in this instance, to receive in the address register, labelled (212), the address, labelled (206), and the memory bank address, labelled (204). In the

timing register, labelled (205), the clock (202), is received along with the chip select command, labelled (201), and the ras, cas, and we commands, labelled (203). The clock signal, delayed by a delay locked loop, labelled (208), is used to drive a data strobe generator, labelled (214), in synchronizing the output data in a read operation through the output buffer, labelled (216), such that the clock and output data are in phase.

The timing register (205) is used to determine if the address bus, labelled (210), is directed to the row buffer, labelled (218), or to the column buffer, labelled (220). A bank activate, or row selection, would occur, for example, if the chip select (201) is low, and the ras is low, and the cas is high of (203); whereas a read or write or column selection, would occur, for example, if chip select (201) is low, and ras, and cas of (203) are both low.

In Figure 3 there is an illustration of a function truth table which indicates the conditions in the DRAM..

Returning to Figure 2, in a condition where the row is selected then the row address is decoded by the row decoder, labelled (222), the 13 address bits would specify one of 8192 possible rows in the data arrays labelled (232). There are 4 possible arrays shown in the Fig. 2 example, the desired one to be activated will be determined by a bank selector, labelled (224).

An important point to be noted with respect to the problems addressed by this invention is that, at this point, all address information, even if some inaccuracy is present, will contain sufficient information to be executed and will and appear to the memory apparatus as being valid. Thus, if there were to be an error on a received address, caused for example by a broken connection in the path between the device creating the address which would be the memory controller of Fig, 6, to be later described, and the memory device, the transmitted data would still be sent or received, and the error could be difficult to detect.

Returning again to Figure 2, once a row has been selected, then the desired column, or columns of that row are

selected, the data in the column buffer is then sent to the column decoder, labelled (226), which selects the required data bits from the final sense amplifiers (234). Since this is a double data rate memory, which in essence means that 2 data bits are to be transferred per DQ port, for every clock cycle, 2 data bits then must be fetched from the arrays 232 for every clock cycle. Which of these 2 bits is output first is determined by the 2 bit column prefetch unit, labelled (236), by inspection of the low order column address CA0, labelled (238). The CA0 (238) is also routed to the data input buffer labelled (242), which receives data in a write operation from the DQ pins, labelled (250), which permits determining that the two bits received on that clock cycle go to the proper address.

The role of CA0(238) is important in the address analysis aspect of the invention in that it is a location where address information is used in the data portion of the chip. In this invention a goal is to bring all of the address and command information, or as much of it as possible, to the data portion of the chip, providing the ability for the information received on the address and command lines to be sent out the data lines, so that a memory controller can check if the address and command it sent to the addressable device, was actually received correctly, by inspection of the data lines. This invention will not affect the contents of the RAM or the normal operation of the addressable device, as the invention will provide a separate bus.

Continuing to refer to figure 2, on a read operation, data from the 2-bit column prefetch unit (236) are driven to the output buffers labelled(216) and out the data ports labelled DQ and (250). At the same time the data strobes UDQS and LDQS at element labelled(214) are driven. The receiving device can use these strobes in the same manner as the RAM to register the data. The timing register controls when the data is driven through the I/O control unit, labelled (252). The data is driven as a burst, unit labelled (262), determines when the burst is over and signals the output buffer. The number of cycles between the read command and the data (the latency) is programmable as is the length of the burst, the programming is done by a memory controller or other external

device by using a special command (mode register write) by proper selection of the commands, cke, cs, ras, cas, and we, that enter the timing register (205); the command itself is contained in the address field. Thus the address path labelled (210) is directed also to the mode register labelled(272), which decodes the instructions during initialization and determines, among other things, the burst length and latency.

This invention makes use of a mode register for memory devices where available and a special command is provided to direct the address bus to the data bus in other words t achieve the address wrap function.

The same address wrap function may be provided in other devices without a mode register by programming.

Returning to Figure 2, similarly if data is being written to the RAM, data received by the input buffer (242) is directed to the selected row and column of the array labelled (232). The data strobes UDQS and LDQS are used to clock in the data, that is, on rising edge of strobe one datum is registered from the receiver and on the falling edge a second datum, hence the double data rate.

With respect to the problem being addressed, again all addresses appear to be valid. If there was an error on a received address, due to example a broken connection in the path between the device creating the address, such as the memory controller, to be discussed in connection with Fig 6, and the memory device, the data to be stored in the memory would still be written, just to the wrong address. This can overwrite otherwise correct data and can have serious consequences in many computers.

It is possible to mask, or block, certain data from being written and in fact on some earlier memory devices certain data could be masked for both a read and a write operation. With this invention, the mask operation is enabled through use of the mask pins, UDM and LDM, labelled (207). UDM masks signals at terminal DQ8-15 and LDM masks signals at terminal DQ0-7, from being written to the array. The operation is through the generation of the iDM signal labelled (282), by the timing register (205), and introduced into the I/O control

(252).

Referring to Figure 3, which is an illustration of a function truth table for a typical in the art SDRAM-DDR, such as illustrated in Fig. 2.and which shows the commands that can be created for the SDRAM-DDR memory devices through use of the commands cke, cs, ras, cas, we, and dm, and the address. The read and write commands have been discussed above, the other commands are not important for this invention but are shown for completeness.

Figure 4 is an illustration of a layout on a subassembly member or card of a two memory bank SDRAM-DDR of the type illustrated in Figs. 1 - 3 with reference numerals added pertaining to this invention. The layout of Fig. 4 is of a typical 128 Megabyte (MB) 184 pin registered 8-byte Dual Inline Memory Module (DIMM) which uses 18, 64 Mb (megabit) SDRAM-DDR devices, labelled (402). The 64 Mb devices are very similar to the standard 256 Mb devices, but with ¼ the capacity and thus 1 fewer address line.will be involved. This DIMM is shown as an example of a collection of addressable devices with a common address bus. There will be many card constructions involving capabilities that will not be involved nor do they affect this invention. Such capabilities are redrive functions labelled (412). There are also data switches labelled (422) shown on the backside, such data switches are used to isolate the memory devices on this DIMM from other DIMMs when multiple DIMMs share a common data bus. The presence or absence of these data switches has no bearing on the address wrap command of this invention. Some card constructions have small EEPROMs labelled (432) which cotatins descriptive information about the DIMM. The presence or absence of this EEPROM has no bearing on our the address wrap command of this invention.

Figure 5 is an illustration showing how a typical two banks of memory devices on front and back of a card as shown in Fig. 4 are wired together. Referring to Figure 5, theSDRAM-DDR devices are labelled (502), the address registers are labelled (512), and the data switches are labelled (522). The address bus to all RAMs is shared. Also the

data bus between the 9 RAMs in the front of the DIMM of figure 4 is shared with the 9 RAMs on the back of the DIMM.

In Figure 6 there is shown a perspective illustration of the arrangement of the SDRAM-DDR type memory cards in a typical in the art computer system as has been discussed in connection with Figures 1-5 with reference numerals added pertaining to this invention.

Referring to Fig. 6 there is shown a high level schematic of the memory and processor sections of a computer system. There is a processor or processors called microprocessors and labelled (610). The processor is connected to a memory controller labelled(620). In some constructions the processor and memory controller may be in the same semiconductor integrated circuit device. There is a service processor labelled (630), for providing such functions as system initialization and error processing. Again the service processor may exist with the processor, or the service processor function can be a part of the processor design. A distinction is made between processor and service processor in this discussion to facilitate explaining data being entered and data that exists in the memory in application of the address wrap function of the invention. The memory controller controls the addressable memory devices labelled (640), which could be SDRAM-DDR memory devices as in figures 1 and 2, in this case shown on memory cards labelled (650) as is typical in the industry at this time, examples of which have been described in connection with Figures 4 and 5. The memory controller (620) receives (reads) or sends (writes) data to the memory through the data bus labelled (660). The controls tell the RAM to read or be written to through the cntrl / address bus labelled (670). While separate data busses and cntrl/address busses, may go to each DIMM , or DRAM or they can be made common, the wrap function of the invention would apply.

It is now common in the industry to have addresses, commands, and data referenced to a clock. The clock can come from a separate chip which synchronizes all components, or, in the case of SDRAM-DDR, or other high speed devices, the practice of sending the clock with the address, command, and data (source synchronous design), is being practiced in the industry. In Fig. 6 the memory controller(620) is shown with separate clock busses labelled (680). How clocking is done is not important for this invention, what is important is that the address wrap, when it occurs, follows the same timing as normal operations.

There is and have been some problems with systems such as those described in Figures 1-6 , which are addressed by the additional command capability provided with this invention.

One problem is that while all addresses to the memory contain valid information but if there is some malfunction in the memory assembly, such as a wiring break, all the information may not be included. Thus it is difficult to know if an address has been properly received. As bus speeds increase, this problem becomes more evident. At the present state of the art, the approach used in testing the memory assembly to determine if addresses are being properly received is by alternatively writing and reading certain data patterns into the RAM at different locations, but this however takes a large number of data transfers to perform and it destroys the contents of the memory. There may be many reasons where it is not desirable to have to destroy the contents of a memory. In accordance with the invention a means of testing the address bus is provided that is fast, and does not destroy the contents of memory.

Another problem is that progress in the art will require signaling techniques that involve adjustments in the timing of when data is launched or captured. In systems such as shown in Figure 6, the speeds of the address and data busses can become very fast, and the detailed timing of the clock with respect to the address / command, and the data strobe with respect to the data becomes more stringent. It would be desirable to be able to periodically

re-initialize such busses.

A further problem may be encountered with high speed DRAMs or other addressable memory assemblies in coming out of a long period of inactivity that timing changes may be encountered due to temperature or voltage variation. It is desirable to have a means of re-establishing timing between the addressable device and the controlling device, without accessing the memory.

In accordance with the invention solutions to these problems are achieved by providing an additional mode function called address wrap, or address echo, which can be accessed under control of a memory controller or a service processor, such as shown in Fig. 6. The purpose of the address wrap or address echo capability of the invention is to provide a bypass of the address information directly to the output where it can be evaluated without interfering with the memory contents.

The following is an illustrative implementation. One way to control the function is to provide a command to the mode register of the addressable device, such as shown as element 272 in Fig.2. Synchronous DRAMs use a mode register, and clock doubled synchronous DRAMs (SDRAM-DDR) have both a mode register, element 272 in Fig. 2, and an extended mode register, known in the art as an (EMR), not separately shown in Fig.2. Address pins A4 are shown in Fig.1. The mode bits A4 in the EMR of SDRAM-DDR are useable when assigned 0 = no address echo mode and 1 = address echo mode. Referring to Figure 7 there is illustrated a bypass circuitry capability within the functional diagram of Fig. 2 that conveys address and data port information with reference numerals pertaining to the invention. In Fig 7 an address serializer labelled (770), takes address information from the address register 212 and conveys it on channel labelled (771) directly to the output buffer 216 and a command serializer labelled (780) takes command information from the timing register 205 and conveys it on channel labelled (781) to the output buffer 216.

Referring to Figure 8 there is illustrated the information flow of the invention in a computer system such as that of Fig. 6 wherein the information path labelled (880) takes a control signal from the memory controller 620 to a selected one of the DRAMs 640 on the card 650 and takes the output address information from the output buffer of the selected DRAM to an evaluation location not shown. Referring to Figure 9 there is illustrated a timing diagram of the performance when the system is in the wrap function or echo function mode of the invention. Addresses and commands on the address / command bus show up several cycles later on the data bus. There are no new signals. On cycle n+2, starting from all DQ pins on the same side of the package, serially readout of the address and control pins located on the same side of the package as presented to the DRAM on cycle n. The order is in ascending pin number for signals located on the side of the package containing pin 1, and is descending pin numbers for the other side of the package. The serial readout should be the same frequency as read data, and obey the same launch and hold time specifications, with the same driver impedance. The serial readout terminates after 8 cycles (16 possible datum), and the data drivers are disabled. One cycle later a new command may be taken. This method is designed to provide the value of the address and control lines all output from the data lines, allowing the controlling device to sense if the sent address (or control) matches the received address (or control). If there is a discrepancy, either the address line is bad, or the data line. The difference can be determined by more complicated functions, for example alternating the order of address readout every other cycle. Other methods are of course possible to achieve different purposes.. For example all address lines latched on cycle n could be output through all data lines on cycle n+2. Similarly there could be different number of cycles (1, 2, 3 etc.) between address latch and data output.

The preferred arrangement for simplicity is to have the timing for the address echo function of the invention be the same as a normal read timing of the memory assembly. This is best for the controller and the RAM, as it preserves the natural timings for normal reads and writes. The address could be serialized, and output as a

serial stream out of one or more data lines. All methods will produce the required information. That is, the value of the address and control lines will be output from the data lines, allowing the controlling device to sense if the sent address (or control) matches the received address (or control). If there is a discrepancy, either the address line is bad, or the data line. The difference can be determined by more complicated functions, for example alternating the order of address readout every other cycle. The EMR can be written at slow speed to improve the chance of a success in the event of a high speed address line problem. To the extent that some of the lines to be tested are required to write the EMR, the test is incomplete. However, failure to enter the address echo mode is itself an indication of address line failure.

An alternative way to invoke the address echo function of the invention is that, referring to figure 3, the SDRAM-DDR function truth table, there is room to define a new command. For example, it could be required that A10, the auto-precharge pin, be used with the burst stop command to create a new command. When A10 is low then the burst stop command acts as before, but when it is high it means the new command, the address wrap or address echo function of this invention. When this new command is given then all addresses and commands are wrapped to the data. .

There are two readily useable applications for this new command technique.

The first is to test address busses. On the memory tester, or in a separate computer system, a copy of the mode register of the device is written and the controller or tester programmed to compare the driven address to the received data. If they do not agree, there is a fault. The test is fast, it does not disturb the contents of memory, and thus it can be performed at any time.

The second is to tune the timing of the assembly to allow high speed operation of address and data busses.

On the memory tester, or in a separate computer system, a copy of the mode register of the device is written and the controller or tester programmed to compare the driven address to the received data. If the data is not received correctly, it might be a broken contact or it might be that the timing relation between address and clock (or data and strobe) is not optimized. For example in a SDRAM-DDR, the controller must align the data strobe which is driven in phase with the data, to the center of the data pulse so as to be insensitive to changes in timing. It is very difficult to know where the center of the data pulse is. The controller can scan the strobe in time with respect to the data, and note when the data fails because the strobe is too early and when the read fails because the strobe is too late. All that is required for this test is that the controller know the data pattern that is to be read. Thus, the address echo or wrap function of the invention allows any data pattern to be sent and then read, in analysis and verification without disturbing the contents of memory.

As the performance of Dynamic Random Access Memories (DRAMS) in data processing systems progresses to ever higher frequencies, precise control of the data input and output in the memory system assembly becomes crucial to ensure that there is reliable transfer into and out of each individual one of the assembly of DRAMs that make up the memory assembly. Included in that precise control is the ability to adjust the impedance of the drivers that move the data in the array. The drivers are separate units known in the art as off chip drivers (OCD)s.

To calibrate the drive strength and impedance of an OCD, DC current measurements can be taken while the OCD is driving a known logical state load and the impedance is adjusted until the required I-V characteristic is obtained. To accomplish such an operation however, the memory controller must be able to establish a desired logical state for the OCD load and then communicate adjustment instructions to the DRAM. The situation is illustrated in connection with Figures 10 - 15 which illustrate the application of the principles of the invention to the control of the impedance of an OCD wherein in Figure 10 there is illustrated a typical DRAM data path for writing data. In Figure 10 a diagram depicts the items essential in the write operation. This particular DRAM has four independent data array banks with the read/write data bus communication channel for the data labelled RWD. The data on the RWD is multiplexed into the arrays.

During a write command , the RW switch places the DRAM in a state to receive and store data. Data is input to the DRAM through the DQ Off-Chip Receivers at a location labelled (OCRs&DQs SYNC) and may be synchronized with a data strobe labelled (DQS). In the event that the architecture is of the prefetch type, where several bits of serial data are latched in parallel on consecutive clock cycles, such data may be reordered if necessary in a multiplexer labelled (WRITE MUX). In either case the data is then driven onto the bidirectional bus labelled (RWD) and is finally stored in the memory array under control of the column control and decoding circuitry.

In Figure 11, there is shown a depiction of the write data path of a DRAM that contains the features of, and operates essentially the same as, the write data path of Figure 10, and in addition there is illustrated, within the dotted bordered section, the features used in providing calibration and control of the impedance of off chip drivers. Referring to Figure 11, an additional control signal labelled (ADJUST) is generated by the DRAM control circuits in response to a mode register set command from the memory controller. When the ADJUST command is active, the RWD bus is disconnected from the data array banks and the write command to the column is suppressed. In other words any data in the memory array will remain undisturbed because the memory array is disabled from accepting and storing data. Therefore, with the ADJUST command active, data can be written onto the RWD bus as with a normal write command, but the data cannot be stored in the memory array. It will be apparent that if at the time the impedance calibration is to be performed, the memory array does not contain data desired to be undisturbed, arrangements to inhibit storage would not be required. The ADJUST command further enables the added control, labelled OCD IMPEDANCE CONTROL, circuitry to receive programming instructions from the data on the RWD bus. The control is clocked using the write command control signal together with a delayed version of the signal. The OCD IMPEDANCE CONTROL element interprets the programming instructions and generates vectors which drive the OCDs and set them to the desired

pull up and pull down levels. An example set of commands and settings are illustrated in Table 1.

Table 1

DQ< 2 >	DQ < 1 >	DQ < 0 >	Command
X	0	0	Do nothing
0	0	1	Increase pulldown impedance
0	1	0	Decrease pulldown impedance
0	1	1	Reset pulldown to default impedance
1	0	1	Increase pullup impedance
1	1	0	Decrease pullup impedance
1	1	1	Reset pullup to default impedance

Thus with the ADJUST signal activated, a normal write command is useable to program the OCD impedance from data on the DQ inputs.

Referring to Figure 12 a timing chart is provided that shows example timings for this operation. The write command signal is labelled PCAS on the chart and the column command is labelled CCAS. The timing chart of Figure 12 assumes the standard in the art write data burst architecture of four bits. Only the first bit of a burst from a subset of n DQs is used for programming information. Alternatively, consecutive bits in a burst could contain programming information.

An example protocol for achieving the impedance adjustment as described in connection with Figure 11 would be as follows.

The extended mode register set activates the ADJUST mode.

The ADJUST mode signal places the RWD MUX in a high impedance mode and disables the write command to the column.

The ADJUST mode signal also prepares the OCD impedance control circuit to receive adjustment instructions.

A single write command captures the DQs and drives them onto the RWDs.

The first bit of the burst on DQ< 0:n > contains the impedance adjustment command. An example command table is Table 1.

Another option is to write the impedance vectors directly to each OCD circuit utilizing the RWD bus to transfer the data to all OCDs and storing the value in a latch at each OCD. This would require that the clocking and mode signals PCAS and ADJUST be distributed to each OCD circuit. Since the existing RWD bus is available to transfer the data to all OCDs the vector bus from the OCD impedance control would no longer be needed thereby saving wiring space.

Figure 13 illustrates an alternate arrangement of adjustment additions to one DQ circuit in a typical data path such as is shown in Fig. 10 in implementing the principles of the invention. Referring to Figure 13, the arrangement does not involve the RWD bus at all and further allows each OCD to be programmed independently. In the arrangement of Fig. 13 two programming mode signals are involved, one labelled (ADJUST _ PU) for adjust OCD pullup and another labelled (ADJUST _ PD) for adjust OCD pulldown. Each can be activated at different times by a mode register set command. When either mode is active the write operation to the array is suppressed as described previously in connection with Fig.11.

During a write command to the DRAM, serial data is received by the off chip receiver(OCR) at each DQ and stored in parallel at the DQ WRITE LATCH . The serial burst length would be four bits. In a usual write command the data would be written in parallel over the RWD bus and into the memory array but the ADJUST_PU or ADJUST_PD mode prevents it. Instead the parallel data is stored directly into the latches located near the OCD. This data contains the value of the desired impedance for either the pullup or pulldown which can then be decoded to select the desired OCD impedance.

Therefore with the ADJUST_PD or ADJUST_PU signal activated a normal write command can be used to program the OCD impedance with the impedance values provided in a serial burst fashion over the DQ inputs. It should be noted that each OCD receives the impedance values from a unique DQ so that independent programming of different OCDs is enabled. It should also further be noted that with this method there is no restriction to a four bit burst length.

An example protocol for achieving the impedance adjustment as described in connection with Figure 13 would be as follows.

The extended mode register set activates the ADJUST _PU or PD mode.

The ADJUST_ PU or PD mode signal places the RWD MUX in a high impedance mode and disables the write command to the column.

A four bit burst is written to each DQ write latch as in a normal write command.

During the DQS to WRTCLK synchronization, the four bit burst is transferred to the pullup or pulldown impedance latch and decoder.

The extended mode register set deactivates the ADJUST_ PUorPD mode signal.

The memory controller performs the impedance measurement.

The procedure is repeated until the adjustment is complete..

Among the applications of the principles of the invention a valuable assistance in the control of the timing relationships is achieved. As the performance of DRAMs is pushed to ever higher frequencies, precise control of the data going in and out of the memory system is crucial to ensuring the reliability of the data. Paramount among the influencing aspects is the ability to adjust for timing skew that develops in the system.

It is desirable to have the ability to place the system in a state where a known stream of data is the output from

the OCDs so that the memory controller can adjust for timing skew using, the standard in the art, vernier type measurement. It is further desirable that the data stream be flexible to allow exercising many different sequences of data. There are situations where prior to calibration it may not be possible to write data to the DRAM so there may not be an option to simply read and write data to the array.

Referring to Figures 14 -16 which illustrate the application of the principles of the invention to the evaluation and control of timing in DRAM memory assemblies. Figure 14 is a depiction of a typical data path in a DRAM assembly. This particular DRAM has four data array banks with the read/write data (RWD) multiplexed onto a common data bus. During a read operation, the signal PCAS is pulsed low while a column address supplied by the memory controller is simultaneously presented on the internal COLADD bus. Within the column access time of the array, the RWD bus will be driven with the data to be output by the OCDs onto the DQ bus. In a prefetch architecture this data is first serialized at the FIFO latches using input and output pointers. If the COLADD bus is not guaranteed until the end of the read cycle, then the starting address must be held until needed to generate the output pointers.

Referring to Figure 15 which illustrates an arrangement of adjustment additions to a data path of the type shown in Fig. 14 in the implementation of the principles of the invention, there is one additional control signal labelled ALIGN that is generated by the DRAM control circuits in response to a mode register set command from the memory controller. When the ALIGN signal is active, the RWD bus is disconnected from the data array banks and connected to the DATA bus via tri-state control. Data is supplied to the DATA bus via the COLADD bus. This allows the RWD bus to be driven with data from the COLADD bus during a normal read operation. The COLADD information is not needed by the column decoder since the data array banks are disconnected from the RWD bus. Therefore any number of consecutive read operations may be performed causing any complex sequence of data to be output by the OCDs onto the DQ bus.

It is noted that the COLADD bus may not be as wide as the RWD bus and therefore a fanout is required to duplicate data across all bits of the bus. Furthermore, the FANOUT function could decode the DATA bus information to produce a variety of complex vectors. One example of this would be to also drive the complement of the DATA bus such that adjacent OCDs would drive complementary data. In addition more Coladd bits could be used to allow more than one unique vector per read operation. Alternatively only a subset of OCDs may be required for calibration and in this case all RWD lines need not be set to a known state.

The depictions of Figs 14 and 15 are illustrative of a synchronous DRAM with a four bit prefetch and a fixed burst length of four bits. In the case where the prefetch is less than the burst length , a counter must be used to issue the data from DATA HOLD at the proper time. It should also be noted that using the starting address DQADD to reorder the burst data in conjunction with a decoding function (e.g.FANOUT) allows the width of the DATA bus to be reduced. For example the six patterns 0000 , 0001 , 0011 , 0101 , 0111 ,and 1111 could be used to produce the ten other possible four bit patterns by simply starting the burst from a different address. Alternatively, DQADD could be forced to a known state by the ALIGN signal and the COLADD bits which are used for the starting address could be used to supply the DATA bus instead.

Referring to Figure 16 a timing chart is provided illustrating the conditions produced in timing where adjustment additions are made in a typical data path such as is illustrated in Fig. 15.

A protocol for vernier type alignment would be as follows:

The extended mode register Set would activate the ALIGN mode signal.

The ALIGN mode signal places the RWD MUX in the high -impedance modeand the driver A takes control of the RWDs.

Any number of normal CAS read commands may follow. In this mode, COLADD < 0,1 > determines the start address and COLADD < 2:5 > provides the data for the four bit burst.

The extended mode register set deactivates the ALIGN mode signal.

There are some observations that may assist in weighing options.

The tristatable drivers A in Fig. 15 can be small since they have the full column to the RWD access time their level.

Address information is latched with an internal CAS command (PCAS).

The FANOUT in Figure 15 is used to distribute the four bit burst to multiple groups of four RWDs.

The COLADD< 0:1> is useable as the first two bits of the four bit burst when the ALIGN mode signal establishes the DQADD<0,1> at a known value.

The COLADD<0:n> is decodable to select one of 2 predefined burst patterns,

FANOUT is useable in generating true and complement data so that adjacent DQs can switch in opposite directions.

More column addresses such as for example COLADD <6:9> are useable to produce more than one unique four bit burst sequences.

What has been described is a control function for a dynamic random access storage array that selectively can bypass the stored memory and permit the address portion of the data to be diverted for analysis, verification and internal control. The control function can be realized with additional structure and by the selective rerouting through existing structure.